Certified Dense Linear System Solving ⋆

T. Mulders

COMIT AG, CH-8004 Zurich, Switzerland

A. Storjohann

School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

Abstract

A randomized algorithm is given for solving a system of linear equations over a principal ideal domain. The algorithm returns a solution vector which has minimal denominator. A certificate of minimality is also computed. A given system has a diophantine solution precisely when the minimal denominator is one. Cost estimates are given for systems over the ring of integers and ring of polynomials with coefficients from a field.

Key words: linear system solution; diophantine system solution; integer matrix; polynomial matrix; randomized algorithm; Las Vegas

1 Introduction

Finding a particular solution to a system of linear equations is a classical mathematical problem. In the literature we typically find separate treatments for two versions of the problem. The first version — *rational system solving* — can be stated as follows: given an integer matrix $A \in \mathbb{Z}^{n \times m}$ and vector $b \in \mathbb{Z}^{n \times 1}$, find a rational vector $x \in \mathbb{Q}^{m \times 1}$ that satisfies $Ax = b$. The second version — *diophantine system solving* — asks for an integer vector x that satisfies $Ax = b$. There are three possibilities:

⋆ The work for this paper was mostly done during both authors' stay at the Institute of Scientific Computing, Department of Computer Science, ETH Zurich, Switzerland

Email addresses: thom.mulders@comit.ch (T. Mulders), astorjoh@scg.uwaterloo.ca (A. Storjohann).

- The system has no rational solution.
- The system has a rational solution but no diophantine solution.
- The system has a diophantine solution.

In this paper we propose a generalization that encompasses all of these situations. Suppose that $Ax = b$ admits a rational solution. If d is the smallest positive integer such that dx is integral, and d is minimal among all solutions to the system, then we call x a solution with *minimal denominator*. We give a randomized algorithm that takes as input an $A \in \mathbb{Z}^{n \times m}$ and $b \in \mathbb{Z}^{n \times 1}$ and returns as output one of the following:

(1) (y, z), where
   - $y \in \mathbb{Q}^{m \times 1}$ with $Ay = b$,
   - $z \in \mathbb{Q}^{1 \times n}$ with $zA \in \mathbb{Z}^{1 \times m}$, and
   - zb and y have the same denominator.

(2) ("no solution", q), where
   - $q \in \mathbb{Q}^{1 \times n}$ with $qA = (0, \ldots, 0) \in \mathbb{Q}^{1 \times m}$ and $qb \neq 0$.

We call this *certified linear system solving*. In the first case, the conditions on y and z certify that y is a solution with minimal denominator. In particular, y is a diophantine solution precisely when the denominator of y is one. In the second case, the existence of such a q certifies that the system has no rational solution. This idea for certifying inconsistency is due to Giesbrecht, Lobo and Saunders (1998).

The main result of this paper is a fast algorithm for certified solving. A complete statement of complexity results, including interaction with fast matrix multiplication, is given in Sections 7.1 and 7.2. Here, we state the results assuming the standard (cubic) algorithm for matrix multiplication.

We show that certified solving of a linear system over $\mathbb{Z}$ can be accomplished using an expected number of $O(nmr\,\mathsf{B}(d + \log m))$ bit operations, where r is the rank of A and d is a bound on the bitlength of entries in A and b. Actually, we show that this complexity bound holds even if entries in b are substantially larger than entries in A. It suffices that d bound both $\log ||A||$ and $(\log ||b||)/r$, where $||A||$ and $||b||$ denote the maximum magnitude of entries in A and b. The function B is a cost function for certain operations with integers and polynomials, see below. The best methods have $\mathsf{B}(t) = O(t(\log t)^2 \log\log t)$.

We obtain an analogous result for an input system over $K[x]$, K a field. Let $A \in K[x]^{n \times m}$ and $b \in K[x]^{n \times 1}$ be given. Certified solving of a linear system over $K[x]$ can be accomplished using an expected number of $O(nmr\,\mathsf{B}(d + \log_{\#K} r))$ field operations from K, where d is a bound for both $||A||$ and $||b||/r$, and $||A||$ and $||b||$ denote the maximum degree of entries in A and b. If K is an infinite field, then $\log_{\#K} r = 0$.

Our algorithms are based on an idea of Giesbrecht (1997). The idea is to construct a diophantine solution of $Ax = b$ by combining a small number of rational solutions of the same system. Giesbrecht computes different rational solutions by solving the leading nonsingular subsystem of $UALx = Ub$, for randomly chosen upper- and lower-triangular toeplitz matrices U and L. Giesbrecht, Lobo and Saunders (1998) extend the algorithm to certify the nonexistence of a diophantine solution, should this be the case. The studies in Giesbrecht (1997) and Giesbrecht et al. (1998) focus on the case of sparse or structured linear systems, with an emphasis also on algorithms which admit a good coarse grain parallelization. If we incorporate the best sequential methods for rational system solving (see Section 5) then the cost of the algorithms there becomes $O(nm^2d^2) \times (\log m + \log d)^{O(1)}$ bit operations, assuming $m \geq n$. The extra logarithmic factors $(\log m + \log d)^{O(1)}$ are due to the rate of convergence and because the proof of convergence requires entries in the toeplitz conditioners to be chosen from a ring extension.

The main technical contributions of the current paper are as follows. First, the idea of certification is extended to verify correctness of a minimal denominator solution. Second, we perform a thorough study of the effectiveness of dense preconditioners, showing how they can be used to avoid the need for extension rings and at the same time improve the rate of convergence to an expected constant number of iterations. The convergence analysis is over a general principal ideal domain and is thus applicable in different settings. Third, we give a complete cost analysis for systems over $\mathbb{Z}$ and $K[x]$. Part of the effort is to show how to incorporate fast arithmetic and matrix multiplication.

We now give a more detailed outline of the rest of the paper.

Sections 2, 3 and 4 study the certified solving problem over an abstract principal ideal domain. Section 2 presents the Algorithm *MinimalSolution* for constructing a solution with minimal denominator together with certificate (y, z) for a full row rank system $Ax = b$. Each iteration of the algorithm constructs a new rational solution by solving the leading nonsingular subsystem of $APx = b$, where P has entries chosen uniformly and randomly from a subset of the ring. Section 3 gives sundry results about the rank properties of random matrices. This section is self-contained and may be of independant interest. Section 4 uses the results of the previous section to estimate the performance of Algorithm *MinimalSolution*. The main result is that we can expect convergence in a constant number of iterations by choosing entries in the preconditioning matrices P from a large enough (but still relatively very small) subset of the ring.

Sections 5, 6 and 7 study the certified solving problem over $\mathbb{Z}$ and $K[x]$. These sections are concerned with efficiency over $\mathbb{Z}$ (expected number of required bit

operations) and over $K[x]$ (expected number of required field operations from K). The algorithms we present work by reducing to the problem of solving a square nonsingular system. Our approach is to bound separately the expected cost of the reduction and the expected number of nonsingular systems that need to be solved. Section 5 gives a brief survey of the currently best known complexity results for solving a nonsingular system over $\mathbb{Z}$ and $F[x]$. Section 6 adapts Algorithm *MinimalSolution* from Section 2 to solve a full row rank system over $\mathbb{Z}$ or $F[x]$. The algorithm from Section 2 needs to be modified slightly to avoid expression swell. Finally, Section 7 gives the Algorithm *CertifiedSolver* for solving a possibly non full row rank and inconsistent system over $\mathbb{Z}$ and $K[x]$.

Cost estimates are given in terms of the subadditive functions M, B and MM.

We assume that $\mathsf{B}(n) = O(n^2)$ or $\mathsf{B}(n) = O(\mathsf{M}(n)\log n)$ where M is a multiplication time for $K[x]$ and $\mathbb{Z}$ (von zur Gathen and Gerhard, 1999, Definition 8.26). Then the extended gcd involving two polynomials from $K[x]$ of degree at most n, or two integers of bitlength at most n, can be computed with $O(\mathsf{B}(n))$ field operations or bit operations, respectively. The best known methods allow $\mathsf{M}(n) = O(n(\log n)(\log\log n))$. We assume that $\mathsf{M}(ab) \leq \mathsf{M}(a)\mathsf{M}(b)$ for $a, b \in \mathbb{Z}_{>1}$. Let MM be such that two $n \times n$ matrices over a ring can can be multiplied in $O(\mathsf{MM}(n))$ ring operations. In this paper we will assume that $n^{2+\gamma} = O(\mathsf{MM}(n))$ for some positive γ.

For a matrix or vector A over $\mathbb{Z}$, we denote by $||A||$ the maximum magnitude of entries in A. For A over $K[x]$, we denote by $||A||$ the maximum degree of entries. Let $\mathcal{L}_{\mathbb{Z}}(n, \alpha, \beta)$ denote the problem of computing $A^{-1}b \in \mathbb{Q}^{n\times 1}$ for a given nonsingular $A \in \mathbb{Z}^{n\times n}$ and $b \in \mathbb{Z}^{n\times 1}$ with $||A|| \leq \alpha$, $||b|| \leq \beta$. Similarly, let $\mathcal{L}_{K[x]}(n, \alpha, \beta)$ denote the problem of computing $A^{-1}b \in K(x)^{n\times 1}$ for a given nonsingular $A \in K[x]^{n\times n}$ and $b \in K[x]^{n\times 1}$ with $||A|| \leq \alpha$, $||b|| \leq \beta$.

2 Certified solving of a consistent system

Let R be a principal ideal domain and F its quotient field. Let $A \in R^{n\times m}$ and $b \in R^{n\times 1}$ be given. Assume throughout this section that the system $Ax = b$ is consistent. This section gives an algorithm to compute a pair (y, z) such that:

- $y \in F^{m\times 1}$ with $Ay = b$.
- $z \in F^{1\times n}$ with $zA \in R^{1\times m}$.
- zb and y have the same denominator.

From these conditions it will follow that y is a solution with minimal denominator. To define precisely what is meant by "denominator" and "minimal

denominator" we need to fix some notation about principal ideal domains. For $v, w \in F$ we say that v and w are associates (notation: $v \sim w$) if there is a unit u in R such that $v = uw$. We assume that for every equivalence class of associate elements we have a unique representative and that this representative is 1 for the class of units in R. In this way we get a unique generator $d(I) \in R$ for every ideal I of R and this allows us to use the term "greatest common divisor" and "least common multiple" without ambiguity.

Definition 1 *Let $x \in F^m$. It is easy to see that the set of all $v \in R$ such that $vx \in R^m$ is an ideal I of R. We denote $d(I)$ by $d(x)$ and call it the* denominator *of x. By $n(x)$ we denote $d(x)x \in R^m$ and call it the* numerator *of x.*

A vector $y \in F^{m \times 1}$ such that $Ay = b$ is called a *rational solution* of the linear system $Ax = b$. If in addition $d(y) = 1$, then y is a *diophantine solution* of the system.

Definition 2 *Let I be the ideal of R generated by the set of denominators of all rational solutions of $Ax = b$. We denote $d(I)$ by $d(A, b)$.*

$d(A, b)$ is the *minimal denominator* that a rational solution of $Ax = b$ can have in the sense that $d(A, b)$ divides $d(y)$ for any rational solution y of $Ax = b$. Clearly, if $Ax = b$ has a diophantine solution, then $d(A, b) = 1$.

The next lemma shows how we can take a linear combination of two rational solutions y_1 and y_2 to produce a new rational solution y with potentially smaller denominator. This idea is due to Giesbrecht (1997).

Lemma 3 *Let $y_1, y_2 \in F^m$ be rational solutions of $Ax = b$. Let $d, s_1, s_2 \in R$ be such that $d = \gcd(d(y_1), d(y_2)) = s_1 d(y_1) + s_2 d(y_2)$. Then*

$$y := \frac{s_1 d(y_1) y_1 + s_2 d(y_2) y_2}{d}$$

is a rational solution of $Ax = b$.

Note that $d(y)$ divides $\gcd(d(y_1), d(y_2))$. From Lemma 3 it follows that a solution with minimal denominator does exist.

Definition 4 *A rational solution y of $Ax = b$ with $d(y) = d(A, b)$ is called a solution with* minimal denominator.

To get different rational solutions of $Ax = b$, we apply the following result for different random choices of P.

Lemma 5 *Let $P \in R^{m \times n}$. If y is a rational solution of $APx = b$, then Py is a rational solution of $Ax = b$.*

By taking linear combinations of several rational solutions as in Lemma 3 we hope to get a sequence of solutions with decreasing and, eventually, minimal denominator. The certification of minimality is based on the next lemma.

Lemma 6 *Suppose $Ax = b$ has a rational solution and let $z \in F^{1\times n}$ such that $zA \in R^{1\times m}$. Then $d(zb)$ divides $d(A, b)$.*

PROOF. Let y be a rational solution of $Ax = b$ with minimal denominator. Then $d(A, b)(zb) = d(A, b)zAy = (zA)(d(A, b)y)$ and $(zA)(d(A, b)y)$ is over R since zA and $d(A, b)y$ are over R. $\square$

Lemma 6 states that z certifies the factor $d(zb)$ of $d(A, b)$. The next lemma shows how we can take a linear combination of two certifying vectors z_1 and z_2 in order to get a new vector z certifying a potentially larger factor of $d(A, b)$.

Lemma 7 *Let $z_1, z_2 \in F^{1\times n}$ such that $z_1A, z_2A \in R^{1\times m}$. Write $z_1b = n_1/d_1$ and $z_2b = n_2/d_2$ where $\gcd(n_1, d_1) = \gcd(n_2, d_2) = 1$. Let $g = \gcd(d_1, d_2)$, $l = \mathrm{lcm}(d_1, d_2)$, $e, s_1, s_2 \in R$ such that*

$$e = \gcd\left(n_1\frac{d_2}{g}, n_2\frac{d_1}{g}\right) = s_1n_1\frac{d_2}{g} + s_2n_2\frac{d_1}{g}.$$

Then $z := s_1z_1 + s_2z_2$ satisfies $zA \in R^{1\times m}$ and $d(zb) = l$.

PROOF. $zA = (s_1z_1 + s_2z_2)A = s_1(z_1A) + s_2(z_2A) \in R^{1\times m}$ and

$$zb = s_1\frac{n_1}{d_1} + s_2\frac{n_2}{d_2} = \frac{s_1n_1d_2 + s_2n_2d_1}{d_1d_2} \sim \frac{s_1n_1d_2 + s_2n_2d_1}{gl} = e/l.$$

Let $p \in R$ be prime. If p divides d_1 but not d_2, then p does not divide n_1d_2 and thus p does not divide e. Similarly, if p divides d_2 but not d_1, then p does not divide e. If p divides both d_1 and d_2, then p does not divide n_1 and n_2. Since also $\gcd(d_1/g, d_2/g) = 1$, p does not divide e. So $\gcd(e, l) = 1$ and thus $d(e/l) = l$. $\square$

To get another $z \in F^{1\times n}$ such that $zA \in R^{1\times m}$, we apply the following lemma for different random choices of P and q.

Lemma 8 *Let $P \in R^{m\times n}$ and $q \in R^{1\times n}$. If $z \in F^{1\times n}$ is such that $z(AP) = q$, then $(d(zA)z)A \in R^{1\times m}$.*

Algorithm *MinimalSolution* is shown in Figure 1. For the input, we assume we have a system $Bx = c$ of full row rank together with a particular rational

algorithm *MinimalSolution*(B, c, y_0)
input: $B \in R^{s \times m}$, $c \in R^{s \times 1}$ and $y_0 \in F^{m \times 1}$, with B of rank s and $By_0 = c$.
comment: The solution y_0 should be from a nonsingular subsystem of $Bx = c$.
output: (y, z), with $y \in F^{m \times 1}$, $z \in F^{1 \times s}$, $By = c$ and $d(y) = d(zc)$.
$U :=$ finite subset of R;
$y := y_0$;
$z := (0, \ldots, 0) \in F^{1 \times s}$;
do
    Choose $P \in U^{m \times s}$ and $q \in U^{1 \times s}$ randomly and uniformly;
    if BP is nonsingular **then**
        $v := (BP)^{-1}c$;
        $\hat{y} := Pv$;
        $y :=$ as in Lemma 3 with $(y_1, y_2) = (y, \hat{y})$;
        $u := q(BP)^{-1}$;
        $\hat{z} := d(uB)u$;
        $z :=$ as in Lemma 7 with $(z_1, z_2) = (z, \hat{z})$;
    fi
until $d(y) = d(zc)$;
return (y, z)

Fig. 1. Algorithm *MinimalSolution*

solution y_0. The general case of a non full row rank system will be reduced to this situation in Section 7. The algorithm takes linear combinations of rational solutions in order to get rational solutions with nonincreasing (and hopefully decreasing) denominator. At the same time linear combinations of certifying vectors are computed in order to get vectors certifying nondecreasing (and hopefully increasing) factors of $d(B, c)$. The loop is iterated until the denominator and certified factor found so far coincide.

The next result follows from the previous lemmas in this section.

Proposition 9 *Algorithm* MinimalSolution *is correct.*

By "correct" we mean that any output produced by the algorithm will be as specified. The next two sections show that we can expect the algorithm to terminate, even if U is chosen to be $\{0, 1\}$.

3 Rank properties of random matrices

We state the results in this section in a general setting so that they can be used in several situations. The coefficients in the matrices we consider are from

a field K. We also use a finite set U and a map $\phi: U \to K$. In this way we cover several possible applications of our results, e.g.

(1) $U \subseteq K$, ϕ the inclusion map.
(2) R a principal ideal domain, U a finite subset of R, $K = R/pR$, where p is a prime in R and ϕ the projection map.

The map ϕ is assumed to be a nonconstant map.

Definition 10 *Let K be a field and A a matrix over K. By* rowSpan(A) *we denote the vector space over K generated by the rows of A. By* colSpan(A) *we denote the vector space generated by the columns.*

The proof of the next result uses counting arguments similar to the analysis in Wiedemann (1986).

Proposition 11 *Let K be a field, $A \in K^{n \times m_1}, B \in K^{n \times m_2}$ and $v \in K^{1 \times m_1}$. Let $t = \operatorname{rank}(A)$ and $s = \operatorname{rank}\left[\begin{array}{cc} A & B \end{array}\right]$. Let U be a finite set and $\phi: U \to K$ a map. Let g be the maximum number of elements in the preimage of any element of K under ϕ. Then*

(a) if $v \notin$ rowSpan(A), then

$$\#\{u \in U^{1 \times m_2} \mid \left[\begin{array}{cc} v & \phi(u) \end{array}\right] \in \operatorname{rowSpan}\left(\left[\begin{array}{cc} A & B \end{array}\right]\right)\} = 0.$$

(b) if $v \in$ rowSpan(A), then

$$\#\{u \in U^{1 \times m_2} \mid \left[\begin{array}{cc} v & \phi(u) \end{array}\right] \in \operatorname{rowSpan}\left(\left[\begin{array}{cc} A & B \end{array}\right]\right)\} \leq (\#U)^{s-t} g^{m_2-(s-t)},$$

with equality when the preimages of all elements of K have the same size.

PROOF. The only nontrivial statement of the proposition is (b). Deleting a row from $\left[\begin{array}{cc} A & B \end{array}\right]$ that is in the row span of the other rows of $\left[\begin{array}{cc} A & B \end{array}\right]$ does not change any essential data in the proposition. Neither does any elementary row operation on $\left[\begin{array}{cc} A & B \end{array}\right]$. So we may assume that $\left[\begin{array}{cc} A & B \end{array}\right]$ has full row rank, i.e. $s = n$, and that $\left[\begin{array}{cc} A & B \end{array}\right]$ is in reduced row echelon form. Let $(j_1, \ldots, j_n)$ be the rank profile of $\left[\begin{array}{cc} A & B \end{array}\right]$. Then $j_t \leq m_1$, $j_{t+1} > m_1$, the first nonzero entry in row i is on the j_i'th position and the j_i'th column is the 0 column, except for

a 1 in the i'th row. A possible configuration for $\left[\begin{array}{c|c} A & B \end{array}\right]$ could look as follows:

$$\left[\begin{array}{cccccccc|cccccc} 1 & * & 0 & * & * & 0 & * & * & 0 & * & 0 & * \\ 0 & 0 & 1 & * & * & 0 & * & * & 0 & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 1 & * & * & 0 & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & * \end{array}\right].$$

Suppose $v \in \mathrm{rowSpan}(A)$. For $u \in U^{1\times m_2}$ we then have:

$$\left[\begin{array}{c|c} v & \phi(u) \end{array}\right] \in \mathrm{rowSpan}\left(\left[\begin{array}{c|c} A & B \end{array}\right]\right)$$

if and only if

$$\begin{array}{l} \text{for all } j \in \{1,\ldots,m_2\} \setminus \{j_{t+1}-m_1,\ldots,j_n-m_1\}: \\ \phi(u_j) \text{ equals the } j\text{th coordinate of} \\ (v_{j_1},\ldots,v_{j_t},\phi(u_{j_{t+1}-m_1}),\ldots,\phi(u_{j_n-m_1}))B. \end{array}$$

So, in order that $\left[\begin{array}{c|c} v & \phi(u) \end{array}\right] \in \mathrm{rowSpan}\left(\left[\begin{array}{c|c} A & B \end{array}\right]\right)$, $u_j \in U$ can be anything for $j \in \{j_{t+1}-m_1,\ldots,j_n-m_1\}$ and they uniquely determine $\phi(u_j)$ for $j \in \{1,\ldots,m_2\} \setminus \{j_{t+1}-m_1,\ldots,j_n-m_1\}$. From this (b) follows easily. □

We remark that, on the one hand, there exist examples where the bound in part (b) of Proposition 11 is sharp. On the other hand, the bound is very pessimistic in many cases. This is because for some choices of the u_j with $j \in \{j_{t+1}-m_1,\ldots,j_n-m_1\}$ there may exist $k \in \{1,\ldots,m_2\} \setminus \{j_{t+1}-m_1,\ldots,j_n-m_1\}$ such that there are less than g different (or even no) $u \in U$ with $\phi(u)$ equal to the kth coordinate of $(v_{j_1},\ldots,v_{j_t},\phi(u_{j_{t+1}-m_1}),\ldots,\phi(u_{j_n-m_1}))B$.

Corollary 12 *When we choose in Proposition 11 the entries in u uniformly from U, then the probability that $\left[\begin{array}{c|c} v & \phi(u) \end{array}\right] \notin \mathrm{rowSpan}\left(\left[\begin{array}{c|c} A & B \end{array}\right]\right)$ is*

$$\begin{cases} 1, & \text{if } v \notin \mathrm{rowSpan}(A); \\ \geq 1-\left(\frac{g}{\#U}\right)^{m_2-(s-t)}, & \text{if } v \in \mathrm{rowSpan}(A), \end{cases}$$

with equality when the preimage of all elements from K have the same size.

We now successively augment rows to a matrix in order to increase its rank. Applying Corollary 12 a number of times gives us a bound for the probability of success.

Lemma 13 *Let K be a field. Let $A \in K^{n_1 \times m_1}$, $B \in K^{n_1 \times m_2}$ and $C \in K^{n_2 \times m_1}$. Let $t = \operatorname{rank}(A)$, $s = \operatorname{rank}\begin{bmatrix} A & B \end{bmatrix}$ and $r = \operatorname{rank}\begin{bmatrix} A \\ C \end{bmatrix}$. Let U be a finite set and $\phi: U \to K$ a map. Let g be the maximum number of elements in the preimage of any element from K under ϕ. Let P be the probability that*

$$\operatorname{rank}\begin{bmatrix} A & B \\ C & \phi(D) \end{bmatrix} = s + n_2,$$

when the entries of $D \in U^{n_2 \times m_2}$ are chosen uniformly from U. Then

$$P \geq \prod_{i=m_2-n_2+r-s+1}^{m_2-(s-t)} \left(1 - \left(\frac{g}{\#U}\right)^i\right),$$

with equality when the preimage of all elements from K have the same size.

PROOF. We choose the rows of D one after the other. Let C_i be the first i rows of C and D_i the first i rows of D. Let $A_i = \begin{bmatrix} A \\ C_i \end{bmatrix}$ and $B_i = \begin{bmatrix} B \\ \phi(D_i) \end{bmatrix}$. Then $\operatorname{rank}\begin{bmatrix} A & B \\ C & \phi(D) \end{bmatrix} = s + n_2$ if and only if $\operatorname{rank}\begin{bmatrix} A_i & B_i \end{bmatrix} = s + i$ for all i, i.e. every row we add must increase the rank by one. Let $t_i = \operatorname{rank}(A_i)$ and $s_i = \operatorname{rank}\begin{bmatrix} A_i & B_i \end{bmatrix}$. Suppose we have chosen D_i such that $s_i = s + i$. Let v_{i+1} be the $(i+1)$'th row of C. We want to choose $u \in U^{1 \times m_2}$ such that $\operatorname{rank}\begin{bmatrix} A_i & B_i \\ v_{i+1} & \phi(u) \end{bmatrix} = s+i+1$, i.e. such that $\begin{bmatrix} v_{i+1} & \phi(u) \end{bmatrix} \notin \operatorname{rowSpan}\left(\begin{bmatrix} A_i & B_i \end{bmatrix}\right)$. Let P_i be the probability that $\begin{bmatrix} v_{i+1} & \phi(u) \end{bmatrix} \notin \operatorname{rowSpan}\left(\begin{bmatrix} A_i & B_i \end{bmatrix}\right)$. From Corollary 12 we get

$$\begin{cases} P_i = 1 & \text{if } v_{i+1} \notin \operatorname{rowSpan}(A_i); \\ P_i \geq 1 - \left(\frac{g}{\#U}\right)^{m_2-(s_i-t_i)} & \text{if } v_{i+1} \in \operatorname{rowSpan}(A_i), \end{cases} \tag{1}$$

with equality when the preimage of all elements from K have the same size.

Since

(a) $t_{i+1} = t_i + 1$, if $v_{i+1} \notin \text{rowSpan}(A_i)$;
(b) $t_{i+1} = t_i$, if $v_{i+1} \in \text{rowSpan}(A_i)$,

we see that case a applies $r - t$ times and that case b applies $n_2 - (r - t)$ times.

If we have chosen u such that $\begin{bmatrix} v_{i+1} & \phi(u) \end{bmatrix} \notin \text{rowSpan}\left(\begin{bmatrix} A_i & B_i \end{bmatrix}\right)$, then $s_{i+1} = s_i + 1$, and so if case (a) applies, then $s_i - t_i$ does not change and if case (b) applies, then $s_i - t_i$ is incremented. Since $P = P_1 P_2 \cdots P_{n_2}$ and $s_0 - t_0 = s - t$, the lemma now follows from (1). □

Definition 14 *Let K be a field and $A \in K^{n \times m}$. We call the set $\{x \in K^m \mid Ax = 0\}$ the* right kernel *of A. $N \in K^{m \times k}$ is called a* right kernel *for A if* $\text{colSpan}(N)$ *is the right kernel of A. In a similar way we define* left kernel.

Lemma 15 *Let K be a field, $A \in K^{n \times m}$ and $B \in K^{m \times k}$. Let N be a right kernel for A. Then*

$$\text{rank}(AB) = \text{rank}\begin{bmatrix} N & B \end{bmatrix} - \text{rank}(N).$$

PROOF. Note that for a matrix M, $\text{rank}(M) = \dim(\text{colSpan}(M))$. Since $\text{colSpan}(AB) = \text{colSpan}\left(A\begin{bmatrix} N & B \end{bmatrix}\right)$, we get:

$$\begin{aligned}
\dim(\text{colSpan}(AB)) &= \dim\left(\text{colSpan}\left(A\begin{bmatrix} N & B \end{bmatrix}\right)\right) \\
&= \dim\left(\text{colSpan}\left(\begin{bmatrix} N & B \end{bmatrix}\right)\right) - \dim\left(\text{colSpan}\left(\begin{bmatrix} N & B \end{bmatrix}\right) \cap \text{colSpan}(N)\right) \\
&= \dim\left(\text{colSpan}\left(\begin{bmatrix} N & B \end{bmatrix}\right)\right) - \dim(\text{colSpan}(N)).
\end{aligned}$$

□

Corollary 16 *Let K be a field, $W_1 \in K^{n \times m_1}$ and $W_2 \in K^{n \times m_2}$ such that $\begin{bmatrix} W_1 & W_2 \end{bmatrix}$ has full row rank, and $M \in K^{m_1 \times n}$. Let $\begin{bmatrix} N_1 \\ N_2 \end{bmatrix}$ be a right kernel for $\begin{bmatrix} W_1 & W_2 \end{bmatrix}$. Let $r_1 = \text{rank}(N_1)$ and $r_2 = \text{rank}\begin{bmatrix} N_1 & M \end{bmatrix}$. Let U be a finite set and $\phi: U \to K$ a map. Let g be the maximum number of elements in the preimage of any element from K under ϕ. When the entries of $P \in U^{m_2 \times n}$*

are chosen uniformly from U, then the probability that

$$\begin{bmatrix} W_1 & W_2 \end{bmatrix} \begin{bmatrix} M \\ \phi(P) \end{bmatrix} \text{ has rank } n,$$

is at least

$$\prod_{i=r_2-m_1+1}^{n+r_1-m_1} \left(1 - \left(\frac{g}{\#U}\right)^i\right),$$

with equality when the preimage of all elements from K have the same size.

PROOF. From Lemma 15 it follows that

$$\operatorname{rank}\left(\begin{bmatrix} W_1 & W_2 \end{bmatrix} \begin{bmatrix} M \\ \phi(P) \end{bmatrix}\right) = \operatorname{rank} \begin{bmatrix} N_1 & M \\ N_2 & \phi(P) \end{bmatrix} - \operatorname{rank} \begin{bmatrix} N_1 \\ N_2 \end{bmatrix}.$$

Using $\operatorname{rank} \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} = m_1+m_2-n$, the lemma now follows by applying Lemma 13 with $A = N_1^t$, $B = N_2^t$, $C = M^t$ and $D = P^t$. $\square$

4 Performance of the Algorithm *MinimalSolution*

We bound the expected number of iterations of the Algorithm *MinimalSolution*. This bound will depend on the size of the set U. If not explicitly stated otherwise, all names represent the variables in the algorithm.

Definition 17 *Let $p \in R$ be prime. For $a \in R$ we define* $\operatorname{ord}_p(a)$ *as the maximum integer n such that p^n divides a.*

In general, $\hat{y}$ in the algorithm will not be a solution of $Bx = c$ with minimal denominator. However, if for a prime $p \in R$ we have $\operatorname{ord}_p(d(\hat{y})) = \operatorname{ord}_p(d(B,c))$ for at least one $\hat{y}$, then the returned solution y will satisfy $\operatorname{ord}_p(d(y)) = \operatorname{ord}_p(d(B,c))$ (Lemma 3.) Similarly, $\hat{z}$ will in general not certify all of $d(B,c)$. However, if for a prime $p \in R$ $\operatorname{ord}_p(d(\hat{z}c)) = \operatorname{ord}_p(d(B,c))$ for at least one $\hat{z}$, then the returned certificate z will satisfy $\operatorname{ord}_p(d(zc)) = \operatorname{ord}_p(d(B,c))$ (Lemma 7.)

Recall that a square matrix V over R is said to be unimodular if V is invertible over R, that is, if V^{-1} is over R. The unimodular matrices over R are precisely those with determinant a unit from R. The following fact is used in the subsequent lemma.

Fact 18 *There exists a unimodular* $V \in R^{m\times m}$ *such that* $BV = H = \left[\begin{array}{cc} H_1 & 0 \end{array}\right]$, *where* H_1 *is* $s \times s$ *and nonsingular. Moreover,* $\det H_1$ *is an associate of the gcd of all* $s \times s$ *minors of* B.

Lemma 19 *Let* $\alpha \in R^m$ *such that* $B\alpha = d(B,c)c$, *that is,* $\alpha/d(B,c)$ *is a solution of* $Bx = c$ *with minimal denominator. Let* V *be as in Fact 18 and* W *the first* s *rows of* V^{-1}. *Let* $p \in R$ *be prime. Let* P *such that* $p \nmid \det(WP)$. *Then* BP *is nonsingular and* $\mathrm{ord}_p(d(\hat{y})) = \mathrm{ord}_p(d(B,c))$. *If moreover* q *is such that* $p \nmid q\det(WP)(WP)^{-1}W\alpha$, *then* $\mathrm{ord}_p(d(\hat{z}c)) = \mathrm{ord}_p(d(B,c))$.

PROOF. Since $B = HV^{-1}$ and $H = \left[\begin{array}{cc} H_1 & 0 \end{array}\right]$ we have

$$B = H_1 W. \tag{2}$$

It follows that $BP = H_1WP$ is nonsingular since H_1 is nonsingular and WP is nonsingular modulo p.

Substituting (2) into $B\alpha/d(B,c) = c$ yields $H_1^{-1}c = W\alpha/d(B,c)$. Then

$$\begin{aligned}\hat{y} &= P(BP)^{-1}c \\ &= P(H_1WP)^{-1}c \\ &= P(WP)^{-1}H_1^{-1}c \\ &= \frac{1}{\det(WP)d(B,c)} \cdot P\det(WP)(WP)^{-1}W\alpha. \end{aligned} \tag{3}$$

From (3) we see that $d(\hat{y})|(\det(WP)d(B,c))$ since $P\det(WP)(WP)^{-1}W\alpha$ is over R. It follows that $\mathrm{ord}_p(d(\hat{y})) \leq \mathrm{ord}_p(d(B,c))$ since by assumption $p \nmid \det(WP)$. On the other hand, we must have $\mathrm{ord}_p(d(B,c)) \leq \mathrm{ord}_p(d(\hat{y}))$ since $\hat{y}$ is a rational solution of $Bx = c$. It follows that $\mathrm{ord}_p(d(\hat{y})) = \mathrm{ord}_p(d(B,c))$.

Since $u = q(BP)^{-1} = q(H_1WP)^{-1} = q(WP)^{-1}H_1^{-1}$ we have

$$\begin{aligned}\hat{z}c &= d(uB)uc \\ &= d(uB)q(WP)^{-1}H_1^{-1}H_1W\alpha/d(B,c) \\ &= \frac{1}{\det(WP)d(B,c)}d(uB)q\det(WP)(WP)^{-1}W\alpha. \end{aligned} \tag{4}$$

Since V is unimodular we have $d(uB) = d(uBV) = d(uH) = d(uH_1)$. Since $p \nmid \det(WP)$, $p|d(uH_1)$ would imply that $p|d(uH_1(WP)) = d(uBP) = d(q) = 1$; a contradiction, so $p \nmid d(uH_1) = d(uB)$. Since $p \nmid q\det(WP)(WP)^{-1}W\alpha$ we see from (4) that $\mathrm{ord}_p(d(\hat{z}c)) \geq \mathrm{ord}_p(d(B,c)$. Since always $\mathrm{ord}_p(d(\hat{z}c)) \leq \mathrm{ord}_p(d(B,c)$ it follows that $\mathrm{ord}_p(d(\hat{z}c)) = \mathrm{ord}_p(d(B,c)$. $\square$

Definition 20 *The pair* (P, q) *is a* good pair *with respect to the prime* p *if*

(1) BP *is nonsingular;*
(2) $\mathrm{ord}_p(d(\hat{y})) = \mathrm{ord}_p(d(B, c))$*;*
(3) $\mathrm{ord}_p(d(\hat{z}c)) = \mathrm{ord}_p(d(B, c))$.

So if we choose in the Algorithm *MinimalSolution* a good pair (P, q) with respect to the prime p, y and z will satisfy from that moment on $\mathrm{ord}_p(d(y)) = \mathrm{ord}_p(d(zc))$.

Lemma 21 *Let* $p \in R$ *be prime,* $\phi: U \to R/pR$ *the projection map and* g *the maximum number of elements in the preimage of any element from* R/pR *under* ϕ*. Then the probability that in a particular iteration of the loop in Algorithm* MinimalSolution *a good pair* (P, q) *with respect to* p *is chosen is at least*

$$\left(1 - \frac{g}{\#U}\right)\left(1 - \frac{g}{\#U} - \left(\frac{g}{\#U}\right)^2\right).$$

PROOF. Let V, W and α be as in Lemma 19. If $p \nmid d(B, c)$ and $p \nmid \det(WP)$ we have for all $q \in U^{1\times s}$ that $\mathrm{ord}_p(d(\hat{z}c)) = \mathrm{ord}_p(d(B, c)) = 0$. So in that case it follows from Lemma 19 that in order for (P, q) to be a good pair with respect to p it suffices that $p \nmid \det(WP)$.

Since V^{-1} is also over R and unimodular, it is clear that W modulo p has rank s. Applying Corollary 16 with $K = R/pR$, $m_1 = 0$ and $W_2 = W$, we see that the probability that $p \nmid \det(WP)$ is at least

$$\prod_{i=1}^{s}\left(1 - \left(\frac{g}{\#U}\right)^i\right).$$

Let $x = g/\#U$. Then

$$\begin{aligned}\prod_{i=1}^{s}(1 - x^i) &\geq \prod_{i=1}^{\infty}(1 - x^i)\\ &= 1 + \sum_{k=0}^{\infty}(-1)^{k+1}\left(x^{(k+1)(3k+2)/2} + x^{(k+1)(3k+4)/2}\right)\\ &\geq 1 - x - x^2.\end{aligned}$$

The second last identity follows from (Hardy and Wright, 1979, Theorem 358). The last inequality uses the observation that for odd k, the sum of the kth and $(k+1)$th term in the sum is positive. The lemma follows when $p \nmid d(B, c)$.

Now assume that $p | d(B, c)$ and $p \nmid \det(WP)$. Suppose $p | W\alpha$. Since the columns of W span all of R^s we then have $(W\alpha)/p = W\beta$ for some $\beta \in R^m$ and

thus $B\beta = H_1 W\beta = H_1 W\alpha/p = B\alpha/p = (d(B,c)/p)c$, contradicting the minimality of $d(B,c)$. So $p \not| W\alpha$ and thus $p \not| \det(WP)(WP)^{-1}W\alpha$. Applying 16 with $K = R/pR$, $m_1 = 0$, $W_2 = (\det(WP)(WP)^{-1}W\alpha)^t$ and $P = q^t$, we see that the probability that $p \not| q\det(WP)(WP)^{-1}W\alpha$ is at least $1 - g/\#U$. The lemma follows from Lemma 19. $\square$

We want the numbers of elements in the preimage of all elements from R/pR under $\phi: U \to R/pR$ to differ as little as possible.

Definition 22 *Let $U \subseteq R$ finite and $p \in R$ prime. We say that U is* evenly distributed *with respect to p, if*

(1) $\#(R/pR) < \infty$*: for all* $w \in R$

$$\left\lfloor \frac{\#U}{\#(R/pR)} \right\rfloor \leq \#\{u \in U \mid u \equiv w \pmod p\} \leq \left\lceil \frac{\#U}{\#(R/pR)} \right\rceil;$$

(2) $\#(R/pR) = \infty$*: for all* $w \in R$

$$\#\{u \in U \mid u \equiv w \pmod p\} \leq 1.$$

Corollary 23 *Let $p \in R$ be prime and U evenly distributed with respect to p. Then the probability that (P,q) is* not *a good pair with respect to p is at most*

$$\begin{cases} \frac{9}{10} & \text{if } \#U = 2 \text{ or } (\#U \geq 25 \text{ and } \#(R/pR) = 2); \\ \frac{2}{\#U} & \text{if } \#U < \#(R/pR); \\ \frac{2}{\#(R/pR)} & \text{if } \#(R/pR) | \#U; \\ \frac{2}{\#(R/pR)} + \frac{2}{\#U} & \text{if } \#(R/pR) \not| \#U. \end{cases}$$

PROOF. Since $(1-x)(1-x-x^2) = 1 - 2x + x^3$ it follows from Lemma 21 that the wanted probability is at most $2g/\#U - (g/\#U)^3 \leq 2g/\#U$. The lemma now follows by noting that

$$g = \begin{cases} 1 & \text{if } \#U < \#(R/pR); \\ \frac{\#U}{\#(R/pR)} & \text{if } \#(R/pR) | \#U; \\ \left\lfloor \frac{\#U}{\#(R/pR)} \right\rfloor + 1 & \text{if } \#(R/pR) \not| \#U. \end{cases}$$

$\square$

One can give sharper bounds for the probability bounded in Corollary 23. However, the bounds in Corollary 23 are easy to use and suffice for our purposes, so we will not give a more detailed analysis of the probability.

Proposition 24 *Let S be a finite set of primes of R. Let $U \subseteq R$ be evenly distributed with respect to all primes in S. For $t \in \mathbb{Z}_{\geq 2}$ and $t = \infty$ let $S_t = \{p \in S \mid \#(R/pR) = t\}$. Then the probability that after N iterations of the loop in Algorithm* MinimalSolution *there is still a prime $p \in S$ such that no good pair (P, q) with respect to p was chosen is at most*

$$\begin{cases} \#S\left(\frac{9}{10}\right)^N & \text{if } \#U = 2; \\ \#S_2\left(\frac{9}{10}\right)^N + \sum_{t>\#U} \#S_t\left(\frac{2}{\#U}\right)^N + \sum_{t|\#U,t>2} \#S_t\left(\frac{2}{t}\right)^N & \\ \qquad + \sum_{t\nmid\#U,2<t<\#U} \#S_t\left(\frac{2}{t}+\frac{2}{\#U}\right)^N & \text{if } \#U \geq 25. \end{cases}$$

PROOF. The wanted probability is at most the sum over all primes $p \in S$ of the probability that no good pair with respect to p was chosen. The probability that N independent experiments, each with a probability of failure less than f, all fail is less than f^N. The lemma now follows from Corollary 23. □

We now apply Proposition 24 when $R = \mathbb{Z}$ and $R = K[x]$. In both cases we will consider U to be a minimal possible set (i.e. $U = \{0, 1\}$) and U of bigger size. Recall that for an integer matrix A we denote by $||A||$ the maximum magnitude of an entry in A. For a polynomial matrix A we denote by $||A||$ the maximum degree of an entry in A. The following well known bounds follow from Cramer's rule and Hadamard's inequality (Horn and Johnson, 1985).

Fact 25 *Let $A \in R^{n\times n}$ be nonsingular, $b \in R^{n\times 1}$ and $y \in F^{n\times 1}$ satisfy $Ay = b$.*

- $(R = \mathbb{Z})$ $d(y) \leq n^{n/2}||A||^n$ *and* $||n(y)|| \leq n^{n/2}||A||^{n-1}||b||$.
- $(R = K[x])$ $\deg d(y) \leq n||A||$ *and* $||n(y)|| \leq (n-1)||A|| + ||b||$.

We will frequently use the following.

Fact 26 *The expected number of experiments one has to perform in order to have success is at most the inverse of a lower bound for the probability that any single experiment has success.*

Corollary 27 $(R = \mathbb{Z})$ *Taking $U = \{0, 1\}$, the expected number of iterations of Algorithm* MinimalSolution *is $O(\log s + \log\log||B||)$.*

PROOF. Let S be the set of prime divisors of the denominator of y_0. By Proposition 24 the probability that after N iterations there is still a prime $p \in S$ such that $\mathrm{ord}_p(d(y)) \neq \mathrm{ord}_p(d(zc))$ is at most $\#S(9/10)^N$. From Fact 26 it then follows that the expected number of iterations in order that $\mathrm{ord}_p(d(y)) =$

$\mathrm{ord}_p(d(zc))$ for all $p \in S$ is at most

$$\frac{N}{\left(1 - \#S\left(\frac{9}{10}\right)^N\right)}. \tag{5}$$

Taking $N = \left\lceil \log_{(10/9)}(2\,\#S) \right\rceil$ we see that (5) is at most $2N$. By Fact 25, $\#S \leq s((\log_2 s)/2 + \log_2 ||B||)$ and the lemma follows. $\square$

Corollary 28 ($R = \mathbb{Z}$) *Taking* $U = \{0, 1, \ldots, M\}$ *where* $M = \max(24, \lceil \log_2 s^{s/2} ||B||^s \rceil)$, *the expected number of iterations of Algorithm* MinimalSolution *is* $O(1)$.

PROOF. The proof is similar to the one of Corollary 27. Note that $\#U \geq \#S + 2$ and $\#U \geq 25$. Now, the probability that after N iterations there is still a prime $p \in S$ such that $\mathrm{ord}_p(d(y)) \neq \mathrm{ord}_p(d(zc))$ is at most

$$\begin{aligned}
\rho &= \left(\frac{9}{10}\right)^N + \sum_{p \in S, p > 2} \left(\frac{2}{p} + \frac{2}{\#U}\right)^N \\
&\leq \left(\frac{9}{10}\right)^N + \sum_{k=3}^{\#S+2} \left(\frac{2}{k} + \frac{2}{\#U}\right)^N \\
&\leq \left(\frac{9}{10}\right)^N + \sum_{k=3}^{A} \left(\frac{2}{k} + \frac{2}{25}\right)^N + \sum_{k=A+1}^{\#S+2} \left(\frac{2}{k} + \frac{2}{\#S+2}\right)^N \\
&\leq \left(\frac{9}{10}\right)^N + \sum_{k=3}^{A} \left(\frac{2}{k} + \frac{2}{25}\right)^N + \sum_{k=A+1}^{\#S+2} \left(\frac{4}{k}\right)^N \\
&\leq \left(\frac{9}{10}\right)^N + \sum_{k=3}^{A} \left(\frac{2}{k} + \frac{2}{25}\right)^N + \sum_{k=A+1}^{\infty} \left(\frac{4}{k}\right)^N,
\end{aligned}$$

and then the expected number of iterations in order that $\mathrm{ord}_p(d(y)) = \mathrm{ord}_p(d(zc))$ for all $p \in S$ is at most $N/(1-\rho)$. Taking $N = 10$ and $A = 10$ we see that this is less than 17. $\square$

Corollary 29 ($R = K[x]$) *Taking* $U = \{0, 1\}$, *the expected number of iterations that Algorithm* MinimalSolution *has to perform is* $O(\log s + \log ||A||)$.

PROOF. Similar to the proof of Corollary 27. Now $\#S \leq s||A||$.

Corollary 30 ($R = K[x]$) *If* K *is not finite, take* $t = 0$ *and* $U \subseteq K$ *to be of size* $\max(25, 3s||B||)$; *if* K *is finite, let* t *be such that* $(\#K)^t \geq 3s||B||$ *and*

take $U = \{f \in K[x] \mid \deg(f) < t\}$. *Then the expected number of iterations of Algorithm* MinimalSolution *is* $O(1)$.

PROOF. Suppose K is not finite. Then $\#U \geq 3(\#S)$ and the probability that after one iteration there is still a prime $p \in S$ such that $\mathrm{ord}_p(d(y)) \neq \mathrm{ord}_p(d(zc))$ is at most $\#S(2/\#U) \leq 2/3$. Thus the expected number of iterations in order that $\mathrm{ord}_p(d(y)) = \mathrm{ord}_p(d(zc))$ for all $p \in S$ is at most 3.

Now suppose that K is finite. There are at most $s||B||/(t+1)$ primes in S of degree $> t$ and at most $(\#K)^k$ primes of degree k. If $\#K > 2$ the probability that after N iterations there is still a prime $p \in S$ such that $\mathrm{ord}_p(d(y)) \neq \mathrm{ord}_p(d(zc))$ is at most

$$\begin{aligned}
\rho &= \frac{s||B||}{t+1}\left(\frac{2}{(\#K)^t}\right)^N + \sum_{k=1}^{t}(\#K)^k\left(\frac{2}{(\#K)^k}\right)^N \\
&\leq \left(\frac{2}{3}\right)^N + \sum_{k=1}^{t} 2\left(\frac{2}{3^k}\right)^{N-1} \\
&\leq \left(\frac{2}{3}\right)^N + \sum_{k=1}^{\infty} 2\left(\frac{2}{3^k}\right)^{N-1},
\end{aligned}$$

and the expected number of iterations in order that $\mathrm{ord}_p(d(y)) = \mathrm{ord}_p(d(zc))$ for all $p \in S$ is at most $N/(1-\rho)$. Taking $N = 8$ this is at most 10.

If $\#K = 2$ there are at most two primes p such that $\#(R/pR) = 2$ and we get

$$\begin{aligned}
\rho &= 2\left(\frac{9}{10}\right)^N + \frac{s||B||}{t+1}\left(\frac{2}{2^t}\right)^N + \sum_{k=2}^{t}(\#K)^k\left(\frac{2}{(\#K)^k}\right)^N \\
&\leq 2\left(\frac{9}{10}\right)^N + \left(\frac{2}{3}\right)^N + \sum_{k=2}^{t} 2\left(\frac{2}{2^k}\right)^{N-1} \\
&\leq 2\left(\frac{9}{10}\right)^N + \left(\frac{2}{3}\right)^N + \sum_{k=1}^{\infty} 2\left(\frac{2}{2^k}\right)^{N-1}.
\end{aligned}$$

The expected number of iterations in order that $\mathrm{ord}_p(d(y)) = \mathrm{ord}_p(d(zc))$ for all $p \in S$ is at most $N/(1-\rho)$. Taking $N = 15$ this is at most 26. □

5 Rational system solving over $\mathbb{Z}$ and $K[x]$

Let $Av = b$ be a nonsingular system of linear equations over R, where $R = \mathbb{Z}$ or $R = K[x]$. The most efficient algorithms for computing $v = A^{-1}b$ are based

on p-adic lifting as described by Moenck and Carter (1979), see also Dixon (1982). The method usually requires knowing a $p \in R$ such that p is relatively prime to $\det A$ (notation: $p \perp \det A$), and p is not a unit of R.

First consider the case $R = \mathbb{Z}$. The complexity analysis of p-adic lifting by Dixon (1982), and by Mulders and Storjohann (1999, Theorem 20), assumes standard integer arithmetic. The incorporation of fast arithmetic is straightforward, but we are not aware of a careful presentation in the literature. We offer a treatment here, indicating only the required modifications to the algorithm as described in (Mulders and Storjohann, 1999).

We are given as input an $A \in \mathbb{Z}^{n \times n}$ and a $b \in \mathbb{Z}^{n \times 1}$. Suppose we are also given a $p \in \mathbb{Z}_{>1}$ such that $p \perp \det A$ and $\log p = O(\log n + \log \alpha)$. Such a p can be chosen at random, as in our algorithms in Sections 6 and 7.

Suppose $||A|| \leq \alpha$ and $||b|| \leq \beta$. Then numerators and denominators in $A^{-1}b$ are bounded in magnitude by $n^{n/2}\alpha^{n-1}\beta$ and $n^{n/2}\alpha^n$, respectively. We will incorporate fast integer multiplication by using the modulus $q = p^k$ instead of p, where k is chosen minimal such that $q^n > 2\lfloor n^{n/2}\alpha^{n-1}\beta \rfloor \lfloor n^{n/2}\alpha^n \rfloor$. Then $\log q = \Theta(\log \alpha + \log n)$ and exactly n steps of q-adic lifting are required to compute the q-adic expansion $A^{-1}b \equiv z_0 + z_1 q + \ldots + z_{n-1}q^{n-1} \bmod q^n$, each $z_* \in \mathbb{Z}^{n \times 1}$ with $||z_*|| < q$.

$B := \mathrm{mod}(A^{-1}, q)$;
$c :=$ the q-adic expansion of b;
comment: Keep c represented as: $c = c_0 + c_1 q + c_2 q^2 + \cdots$
for i **from** 0 **to** $n-1$ **do**
    $z_i := \mathrm{mod}(Bc_i, q)$;
    $c := c - Az_i q^i$
od;

The inverse B can be computed with $O(n^3\,\mathsf{M}(\log q) + n^2\,\mathsf{B}(\log q))$ bit operations by working over $\mathbb{Z}/(q)$, see for example Storjohann (2000, page 55). The reason for the $n^2\,\mathsf{B}(\log q)$ term is that $O(n^2)$ gcd-type operations may be required since q is not necessarily a prime. After stage i of the loop, $A^{-1}b = z_0 + z_1 q + \cdots + z_i q^i + A^{-1}c$, where c is divisible by q^i. It follows that $c = b - A(z_0 + z_1 q + \cdots + z_i q^i)$, which shows that $\log c = O(n \log q)$ throughout. The key to performing the lifting efficiently is to keep c in q-adic representation. The initial expansion of a single entry of b can be accomplished with $O(\mathsf{B}(n \log q))$ bit operations using radix conversion (von zur Gathen and Gerhard, 1999, Section 9.2). A cost bound of $O(n^2\,\mathsf{M}(\log q))$ bit operations for a single iteration of the loop is now easily obtained. After the code fragment finishes, compute $z := z_0 + z_1 q + \cdots + z_{n-1}q^{n-1}$ by applying radix conversion to each entry. Finally, Wang and Pan (2003) prove that rational reconstruction can be applied to an entry of z

at a cost of $O(\mathsf{B}(n \log q))$ bit operations. Note that $O(\mathsf{B}(n \log q))$ is bounded by $O(n^2\, \mathsf{B}(\log q))$, using the simplification $\mathsf{B}(n \log q) = O(\mathsf{B}(n)\, \mathsf{B}(\log q))$, and then $\mathsf{B}(n) = O(n^2)$.

This variation of p-adic lifting described above supports the running time bounds in Proposition 31. Part 1 of the proposition, the analysis in terms of α and β, was already given by Mulders and Storjohann (1999, Theorem 20).

Proposition 31 (Cost of $\mathcal{L}_{\mathbb{Z}}(n, \alpha, \beta)$) *Let nonsingular $A \in \mathbb{Z}^{n \times n}$ and $b \in \mathbb{Z}^{n \times 1}$ be given, $||A|| \leq \alpha$, $||b|| \leq \beta$. Then $A^{-1}b \in \mathbb{Q}^{n \times 1}$ can be computed with*

(1) $O(n^3(\log \alpha + \log n)^2 + n(\log \beta)^2)$ *bit operations using standard integer arithmetic, assuming we are given a $p \in \mathbb{Z}_{>1}$ such that $p \perp \det A$ and $\log p = O(\log \alpha + \log n)$.*
(2) $O(n^3\, \mathsf{B}(\log \alpha + \log n))$ *bit operations, assuming $\log \beta = O(n \log \alpha)$ and we are given a $p \in \mathbb{Z}_{>1}$ such that $p \perp \det A$ and $\log p = O(\log \alpha + \log n)$.*

Now consider the case $R = K[x]$. The construction of algorithms over $K[x]$ is considerably easier than over $\mathbb{Z}$ because the degree norm for polynomials is non-Archimedian (we don't have a problem with carries). Some improved results are available. The first result of Proposition 32 is obtained using an algorithm by Mulders and Storjohann (2000, Theorem 3). That algorithm allows performing the lifting with the modulus x^α, even when x divides $\det A$. The second result, incorporating matrix multiplication into the lifting algorithm, is due to Storjohann (2003, Corollary 16).

Proposition 32 (Cost of $\mathcal{L}_{K[x]}(n, \alpha, \beta)$) *Let nonsingular $A \in K[x]^{n \times n}$ and $b \in K[x]^{n \times 1}$ be given, $||A|| \leq \alpha$, $||b|| \leq \beta$. Then $A^{-1}b \in K(x)^{n \times 1}$ can be computed with*

(1) $O(n^3\, \mathsf{M}(\alpha) + n^2\beta/\alpha\, \mathsf{M}(\alpha) + n\, \mathsf{B}(n\alpha + \beta))$ *field operations.*
(2) $O(\mathsf{MM}(n)(\log n)\, \mathsf{M}(\alpha + \deg p) + \mathsf{MM}(n)\, \mathsf{B}(\alpha + \deg p) + n\, \mathsf{B}(n(\alpha + \deg p)))$ *field operations, assuming $\beta = O(n\alpha)$ and we are given a nonconstant $p \in K[x]$ such that $p \perp \det A$.*

Note that the bound in part 1 of Proposition 32 simplifies to $O(n^3\, \mathsf{M}(\alpha) + n\, \mathsf{B}(n\alpha))$ field operations if $\beta = O(n\alpha)$. The bound in part 2 simplifies to $O(\mathsf{MM}(n)(\log n)\mathsf{B}(\alpha + \deg p))$ field operations under the additional assumption that $\mathsf{B}(t) = O(\mathsf{MM}(t)/t)$. This assumption on $\mathsf{B}(t)$ stipulates that if fast matrix multiplication techniques are used, then fast polynomial arithmetic should be used also.

6 Certified solving of a consistent system over $\mathbb{Z}$ and $K[x]$

We give a modification of Algorithm *MinimalSolution* that is suited to the case when $R = \mathbb{Z}$ or $R = K[x]$. We first explain the required modifications, present the algorithm, and then estimate the complexity in each of these cases.

To avoid expression swell, we need to change how the various rational solutions and certificates are combined. We use the following two lemmas. Correctness of the first lemma is easy. The proof of the second lemma is similar to that of Lemma 7.

Let $A \in R^{n \times m}$ and $b \in R^{n \times 1}$.

Lemma 33 *Let $y_0, y_1, y_2 \in F^{m \times 1}$ be rational solutions of $Ax = b$. Let $a \in R$ be such that $\gcd(d(y_0), d(y_1) + ad(y_2)) = \gcd(d(y_0), d(y_1), d(y_2))$. Then*

$$y := \frac{d(y_1)y_1 + ad(y_2)y_2}{d(y_1) + ad(y_2)}$$

is a rational solution of $Ax = b$ and $\gcd(d(y_0), d(y))$ divides $\gcd(d(y_0), d(y_1), d(y_2))$.

Lemma 34 *Let $z_1, z_2 \in F^{1 \times n}$ such that $z_1A, z_2A \in R^{1 \times m}$. Write $z_1b = n_1/d_2$ and $z_2b = n_2/d_2$, where $\gcd(n_1, d_1) = \gcd(n_2, d_2) = 1$. Let $g = \gcd(d_1, d_2)$ and $l = \operatorname{lcm}(d_1, d_2)$. Then $\gcd(n_1d_2/g, n_2d_1/g, l) = 1$. Let $a \in R$ be such that $\gcd(n_1d_2/g + an_2d_1/g, l) = 1$. Then $z := z_1 + az_2$ satisfies $zA \in R^{1 \times m}$ and $d(zb) = l$.*

Figure 2 gives a detailed description of the modified algorithm. In order to keep y and z small, we only combine them with new solutions or certificates when this will lead to some progress in the computation, i.e. $d(y)$ gets smaller or $d(zc)$ gets bigger.

For T we will choose a set of primes such that for nonsingular BP, $BP \bmod p$ is singular over $R/(p)$ for at most half of the primes $p \in T$. When p is well chosen, one iteration of Algorithm *SpecialMinimalSolution* is similar to one iteration of Algorithm *MinimalSolution*. The next result now follows from Fact 26 and the previous lemmas in this section.

Proposition 35 *Algorithm* SpecialMinimalSolution *is correct. The expected number of iterations of the algorithm is at most two times the expected number of iterations of Algorithm* MinimalSolution.

algorithm *SpecialMinimalSolution*(B, c, y_0)
input: $B \in R^{s \times m}$, $c \in R^n$ and $y_0 \in R^{m \times 1}$, with B of rank s and $By_0 = c$.
comment: The solution y_o should be from a nonsingular subsystem of $Bx = c$.
output: (y, z), with $y \in R^{m \times 1}$, $z \in R^{1 \times s}$, $By = c$ and $d(y) = d(zc)$.
$U :=$ finite subset of R;
$T :=$ SetOfPrimes(B, U);
$y := y_0$;
$z := (0, \ldots, 0) \in R^{1 \times s}$;
do
    Choose $P \in U^{m \times s}$ and $p \in T$ randomly and uniformly;
    if BP mod p is nonsingular **then**
        $v := (BP)^{-1}c$;
        $\hat{y} := Pv$;
        if $\gcd(d(y_0), d(y), d(\hat{y})) \neq \gcd(d(y_0), d(y))$ **then**
            $y :=$ as in Lemma 33 with $(y_0, y_1, y_2) = (y_0, y, \hat{y})$
        fi;
        Choose $q \in U^{1 \times s}$ randomly and uniformly;
        $u := q(BP)^{-1}$;
        $\hat{z} := d(uB)u$;
        if $\mathrm{lcm}(d(zc), d(\hat{z}c)) \neq d(zc)$ **then**
            $z :=$ as in Lemma 34 with $(z_1, z_2) = (z, \hat{z})$
        fi
    fi
until $\gcd(d(y_0), d(y)) = d(zc)$;
$y :=$ as in Lemma 3 with $(y_1, y_2) = (y, y_0)$;
return (y, z)

Fig. 2. Algorithm *SpecialMinimalSolution*

6.1 Complexity when $R = \mathbb{Z}$

Most of our effort is to bound the bitlengths of numbers occurring during the algorithm. When the elements in U are bounded in magnitude by M, then $||BP|| \leq mM||B||$ and $\det BP$ is bounded in magnitude by $N = (s^{1/2}mM||B||)^s$. Let $l = 6 + \lceil \log \log N \rceil$ and choose T to be a set of $2\lceil \lceil (\log_2 N \rceil / (l-1) \rceil$ primes between 2^{l-1} and 2^l. Giesbrecht (1993, Theorem 1.8, based on bounds by Rosser and Schoenfeld (1962)) shows that there are at least this many primes in this range and notes that the construction of T can be accomplished with $O(\log N \log \log \log N)$ bit operations using the sieve of Erastosthenese, see (Knuth, 1981, Section 4.5.4).

In what follows we will either take $U = \{0, 1\}$ or take $U = \{0, 1, \ldots, M\}$, where $M = \max(24, \lceil \log_2(s^{s/2}||B||^s) \rceil)$. It follows that primes in T have bitlength bounded by $O(\log s + \log \log m + \log \log ||B||)$; we use this bound implicitly in what follows. By Fact 25, the following bitlength bounds hold throughout

execution of the algorithm:

$n(y_0)$	$O(s(\log s + \log \|\|B\|\|) + \log \|c\|\|)$
$d(y_0)$	$O(s(\log s + \log \|\|B\|\|))$
$d(v), d(\hat{y}), d(u), n(u), n(\hat{z})$	$O(s(\log m + \log \|\|B\|\|))$
$n(v), n(\hat{y})$	$O(s(\log m + \log \|\|B\|\|) + \log \|\|c\|\|)$

Let V, H and H_1 be as in Fact 18. Since $\hat{z}B \in R^{1\times m}$ we also have $\hat{z}\begin{bmatrix} H_1 & 0 \end{bmatrix} = \hat{z}BV \in R^{1\times m}$ and thus $d(\hat{z})|\det(H_1)$. In the same way we find $d(z)|\det(H_1)$. Since $\gcd(d(y_0), d(y))$ and $d(zc)$ are always bounded by $d(y_0)$ it follows that y and z will be modified at most $O(s(\log s + \log ||B||))$ times. The a of Lemmas 33 and 34 will be computed to have magnitude bounded by $d(y_0)$ and $\operatorname{lcm}(d(z), d(\hat{z}))$ respectively. This gives the following length bounds holding throughout execution of the algorithm:

$d(\hat{z}), d(z)$	$O(s(\log s + \log \|\|B\|\|))$
$n(y)$	$O(s(\log m + \log \|\|B\|\|) + \log \|\|c\|\|)$
$d(y), n(z)$	$O(s(\log m + \log \|\|B\|\|))$

We get the following lemmas.

Lemma 36 ($R = \mathbb{Z}$) *Let (y, z) be output from Algorithm* SpecialMinimalSolution. *Then $d(y)$ and $d(z)$ have bitlength bounded by $O(s(\log ||B|| + \log s))$. Entries of $n(y)$ and $n(z)$ have bitlength bounded by $O(s(\log ||B|| + \log m) + \log ||c||)$ and $O(s(\log ||B|| + \log m))$ respectively.*

Lemma 37 ($R = \mathbb{Z}$) *Assume that $\log ||U|| = O(\log s + \log\log ||B||)$. The cost of one iteration of the loop in Algorithm* SpecialMinimalSolution, *except for the computation of v and u, is bounded by $O(m(\mathsf{MM}(s)/s)\,\mathsf{M}(d + \log m) + m\mathsf{B}(s(d + \log m)))$ bit operations, where d is a bound for both $\log ||B||$ and $(\log ||c||)/s$.*

PROOF. Integers throughout are bounded in length by $O(s(d+\log m))$ bits. For most of the steps (eg. computing denominators, gcds, lcms, vector arithmetic, computation of BP etc.) the lemma now follows from standard complexity considerations.

For the computation of a in Lemmas 33 and 34 we can use an algorithm described in (Mulders and Storjohann, 1999) when $\mathsf{B}(n) = O(n^2)$ and in (Storjohann and Mulders, 1998) when $\mathsf{B}(n) = O(\mathsf{M}(n)\log n)$.

For the computation of Pv, proceed as follows.

(1) Divide the entries in $n(v)$ in chunks of length $\lceil(\log_2 ||n(v)||)/s\rceil$ bits and consider v as an $s \times O(s)$ matrix V. Note that $\log ||V||| = O(d + \log m)$.
(2) Compute $d(v)Pv$ from PV by shifts and additions.

This shows that Pv can be computed in the allotted time. The computation of uB is accomplished similarly. □

The next result follows immediately from Lemma 37, Proposition 35 and Corollaries 27 and 28.

Proposition 38 ($R = \mathbb{Z}$) *Let d be a bound for both $\log ||B||$ and $(\log ||c||)/s$.*

- *Taking $U = \{0, 1, \ldots, M\}$ with $M = \max(24, \lceil \log_2 s^{s/2}||B||^s\rceil)$, the expected cost of Algorithm* SpecialMinimalSolution *is $O(m(\mathsf{MM}(s)/s)\,\mathsf{M}(d+\log m)+ m\mathsf{B}(s(d + \log m)))$ bit operations, plus the cost of solving an expected $O(1)$ instances of $\mathcal{L}_{\mathbb{Z}}(s, mM||B||, \max(M, ||c||))$.*
- *Taking $U = \{0, 1\}$, the expected cost is $O((m(\mathsf{MM}(s)/s)\,\mathsf{M}(d + \log m) + m\mathsf{B}(s(d + \log m))) \cdot (\log s + \log\log ||B||))$ bit operations, plus the cost of solving an expected $O(\log s + \log\log ||B||)$ instances of $\mathcal{L}_{\mathbb{Z}}(s, ||B||, ||c||)$.*

6.2 Complexity when $R = K[x]$

When the elements in U have degree bounded by t, the entries in BP have degree bounded by $||B||+t$ and thus the degree of any minor of BP is bounded by $N := s(||B|| + t)$. Choose T in step (1) differently depending on the size of K.

(Case 1: $\#K \geq 2N$) Take for T a set of $2N$ polynomials of the form $X - a$, with $a \in K$.

(Case 2: $\#K < 2N$) Let $q = \#K$ and let $l \in \mathbb{Z}$ be minimal such that $q^l - q(q^{l/2} - 1)/(q - 1) \geq 2N$. Then $l = O(\log_q N)$. From Lidl and Niederreiter (1983, Exercise 3.27) it follows that there are $\geq 2N/l$ monic irreducible polynomials of degree l over K. Thus, we can take for T the set of all monic irreducible polynomials of degree l. The explicit construction of T is not actually required. A random irreducible polynomial of degree l (chosen randomly and uniformly from the set of all such polynomials) can be constructed with an expected number of $O(l^3)$ field operations, see for example Shoup (1994).

In what follows we will either take $U = \{0, 1\}$ or take U as in Corollary 30. Then $t = O(\log_q s + \log_q ||B||)$ and it follows that the degree of a prime in

T is bounded by $O(\log_q s + \log_q ||B||)$. Similar to the integer case, we get the following degree bounds holding throughout the algorithm:

$n(y_0)$	$O(s\|\|B\|\| + \|\|c\|\|)$
$d(y_0), d(\hat{z}), d(z)$	$O(s\|\|B\|\|)$
$d(u), d(\hat{y}), d(v), n(v), n(\hat{z}), d(y), n(z)$	$O(s(\|\|B\|\| + t))$
$n(u), n(\hat{y}), n(y)$	$O(s(\|\|B\|\| + t) + \|\|c\|\|)$

We get the following lemmas.

Lemma 39 ($R = K[x]$) *Let (y, z) be output from Algorithm* SpecialMinimalSolution. *Then $d(y)$ and $d(z)$ have degree bouded by $O(s||B||)$. Let t be the maximum degree of entries in U. Then entries of $n(y)$ and $n(z)$ have degree bounded by $O(s(||B|| + t) + ||c||)$ respectively.*

Lemma 40 ($R = K[x]$) *Let t be the maximum degree of entries in U. The cost of one iteration of the loop in Algorithm* SpecialMinimalSolution*, expect for computation of v and u, is bounded by $O(m\,(\mathsf{MM}(s)/s)\,\mathsf{M}(d+t)+m\mathsf{B}(s(d+t)))$ field operations, where d is a bound for both $||A||$ and $||c||/s$.*

The proof of Lemma 40 is analogous to the proof of Lemma 37.

The next result follows immediately from Lemma 40, Proposition 35 and Corollaries 29 and 30.

Proposition 41 ($R = K[x]$) *Let d be a bound for both $||B||$ and $||c||/s$.*

- *Taking U as in Corollary 30, the expected cost of Algorithm* SpecialMinimalSolution *is $O(m(\mathsf{MM}(s)/s)\,\mathsf{M}(d+t)+m\mathsf{B}(s(d+t)))$ field operations, plus the cost of solving $O(1)$ instances of $\mathcal{L}_{K[x]}(s, ||B|| + t, \max(t, ||c||))$, where $t = 0$ if $\#K$ is infinite and $t = \lfloor \log_{\#K}(3s||B||) \rfloor$ otherwise.*
- *Taking $U = \{0, 1\}$, the expected cost is bounded by $O((m(\mathsf{MM}(s)/s)\,\mathsf{M}(d) + m\mathsf{B}(sd))\cdot(\log s + \log ||B||))$ field operations, plus the cost of solving $O(\log s + \log ||B||)$ instances of $\mathcal{L}_{K[x]}(s, ||B||, ||c||)$.*

7 Certified solving over $\mathbb{Z}$ and $K[x]$

Let $R = \mathbb{Z}$ or $R = K[x]$. This section presents an extension of Algorithm *SpecialMinimalSolution* that solves the certified linear system solving problem. The algorithm that takes as input an $A \in R^{n\times m}$, which may not be of full row rank, together with a $b \in R^{n\times 1}$ and returns as output one of the following:

(1) (y, z), where
- $y \in F^{m\times 1}$ with $Ay = b$,
- $z \in F^{1\times n}$ with $zA \in R^{1\times m}$, and
- zb and y have the same denominator.

(2) ("no solution", q), where
- $q \in F^{1\times n}$ with $qA = (0, \ldots, 0) \in F^{1\times m}$ and $qb \neq 0$.

The idea of certifying inconsistency as in 2. is due to Giesbrecht et al. (1998, Theorem 2.1), who make the following observation.

Theorem 42 *Let $A \in F^{n\times m}$ and $b \in F^{n\times 1}$. There is no $x \in F^{m\times 1}$ such that $Ax = b$ if and only if there exists a $u \in F^{1\times n}$ such that $uA = (0, \ldots, 0) \in F^{1\times m}$ and $ub \neq 0$.*

Algorithm *CertifiedSolver* is shown in Figure 3. The algorithm is an easy extension of Algorithm *SpecialMinimalSolution*. Let $r = \text{rank}[A]$ and $\bar{r} = \text{rank}[A \,|\, b]$. Then $r \leq \bar{r} \leq r+1$ and the system $Ax = b$ is inconsistent precisely when $\bar{r} = r+1$. This test for inconsistency is performed in step 2 by computing the rank over $R/(p)$ for a randomly chosen prime p. The set T will be chosen so that for at least half the primes $p \in T$ we have, in step 2, that $s = \text{rank}[A]$. After step 2 and throughout the algorithm we will always have $s \leq \bar{s} \leq s+1$, $s \leq r$ and $\bar{s} \leq \bar{r}$. Now consider step 3. Assume, without loss of generality, that $P = I_n$ and $Q = I_m$.

In case $s = \bar{s}$, Algorithm *SpecialMinimalSolution* is used to compute a minimal denominator solution y to the full row rank subsystem $[A_{11} \,|\, A_{12}]x = b_1$. The algorithm then checks if y is a solution to the entire system $Ax = b$. Note that if $s = r$ and $r = \bar{r}$ then this check will not fail. If this check does fail, then we know that $\bar{s} < \bar{r}$, so we increment $\bar{s}$, adjust P as indicated and return to step 3 with $\bar{s} = s+1$. Note that if we are arriving at step 3 from step 4, then the first $\bar{s}$ rows of $P[AQ|b]$ have rank only $\bar{s}-1$ over $R/(p)$, but rank $\bar{s}$ over R, as required.

In case $\bar{s} = s+1$, the algorithm attempts to compute a certificate for inconsistency. By construction,

$$\left[\begin{array}{c|c} u & -1 \end{array}\right] \left[\begin{array}{cc|c} A_{11} & A_{12} & b_1 \\ \hline A_{21} & A_{22} & b_2 \end{array}\right] = \left[\begin{array}{cc|c} & & b_1 \\ \hline & c & \bullet \end{array}\right] \tag{6}$$

where $c \bmod p$ is zero and $\bullet$ is nonzero when c is zero. The algorithm then checks if c is zero over R, in which case the system is certified to be inconsistent. Note that if $s = r$ then this check will not fail.

algorithm *CertifiedSolver*
input: $A \in R^{n\times m}$, $b \in R^{n\times 1}$.
output: Either (y, z) with $y \in F^{m\times 1}$, $z \in F^{1\times n}$, $Ay = b$, $zA \in R^{1\times m}$ and $d(y) = d(zA)$ OR ("no solution", q) where $q \in F^{1\times n}$ such that $qA = 0$ and $qb \neq 0$.

(1) $T :=$ SetOfPrimes(A);
(2) Choose p randomly and uniformly from T;
$s :=$ rank$(A \bmod p)$;
$\bar{s} :=$ rank$([A|b] \bmod p)$;
$P, Q :=$ permutation matrices as indicated below;
(3) Write $P[AQ|b]$ using a block decomposition as

$$P[AQ|b] = \left[\begin{array}{cc|c} A_{11} & A_{12} & b_1 \\ A_{21} & A_{22} & b_2 \\ \hline A_{31} & A_{32} & b_3 \end{array}\right],$$

where A_{11} is $s \times s$ with rank s, A_{21} is $(\bar{s} - s) \times s$ and the first $\bar{s}$ rows of the augmented system have rank $\bar{s}$.
(4) **if** $s = \bar{s}$ **then**
comment: $\left[\begin{array}{cc|c} A_{21} & A_{22} & b_2 \end{array}\right]$ has dimension $0 \times (m + 1)$.
$v := A_{11}^{-1} b_1$;
$y_0 := (v, 0, \ldots, 0) \in F^{n\times 1}$;
$(y, z) :=$ SpecialMinimalSolution$([A_{11}|A_{12}], b_1, y_0)$;
if $[A_{31}|A_{32}]y \neq b_3$ **then**
Let i be such that i'th entry of $[A_{31}|A_{32}]y - b_3$ is nonzero;
Interchange row $s + 1$ and $s + i$ of P;
$\bar{s} := \bar{s} + 1$;
goto (3)
fi;
$z := (z, 0, \ldots, 0) \in F^{1\times n}$;
return (Qy, zP)
else
$u := A_{21} A_{11}^{-1}$;
if $uA_{12} \neq A_{22}$ **then goto** (2) **fi**;
$q := (u, -1, 0, \ldots, 0) \in F^{1\times m}$;
return ("no solution", qP)
fi

Fig. 3. Algorithm *CertifiedSolver*

Recall that T is chosen so that at most half the primes in T cause repetition of the algorithm. The next result now follows from Fact 26.

Proposition 43 *Algorithm* CertifiedSolver *is correct. The algorithm repeats step 2 an expected number of fewer than two times.*

7.1 Complexity when $R = \mathbb{Z}$

A maximal rank minor of A is bounded in dimension by m and hence in magnitude by $N = (m^{1/2}||A||)^m$. As explained in §6.1, we can set $l = 6 + \lceil \log\log N \rceil$ and choose T to be a set of $2\lceil \lceil \log_2 N \rceil / (l-1) \rceil$ primes between 2^{l-1} and 2^l. Then primes in T have length bounded by $O(\log m + \log\log ||A||)$ bits.

Proposition 44 ($R = \mathbb{Z}$) *The expected cost of Algorithm* CertifiedSolver *is bounded $O(nm(\mathsf{MM}(r)/r^2)\mathsf{M}(d + \log m) + (n+m)\,\mathsf{B}(r(d+\log m)))$ bit operations, where r is the rank of A and d is bound for both $\log||A||$ and $(\log||b||)/r$, plus the cost of solving an expected $O(1)$ instances of $\mathcal{L}_{\mathbb{Z}}(r, mM||A||, \max(M, ||A||, ||b||))$, where $M = \max(24, \lceil \log_2 r^{r/2}||A||^r \rceil)$.*

PROOF. The cost of computing $[A \mid b] \bmod p$ is bounded by $O(nm\,\mathsf{M}(\log||A||) + n\mathsf{M}(\log||B||))$. The ranks $(s, \bar{s})$ over $R/(p)$ are recovered by computing a row echelon form of $[A \mid b] \bmod p$. This costs $O(nm\,(\mathsf{MM}(r)/r^2)\,\mathsf{M}(\log p) + r\,\mathsf{B}(\log p))$ bit operations using an algorithm of Storjohann and Mulders (1998), see also (Storjohann, 2000, Chapter 2). At the same time we can recover permutation matrices P and Q such that the principal $s \times s$ submatrix of PAQ is nonsingular modulo p and the first $\bar{s}$ rows of $P[A|b]$ are linearly independant over $R/(p)$. This shows that step 2 can be accomplished in the allotted time.

Now consider step 4. Lemma 36 bounds $\log||n(y)||$ by $O(r(\log m + \log||A||) + \log||b||)$. For the computation of $[A_{31}|A_{32}]y$ use the same technique as used to compute Pv in the proof of Proposition 37. Compute uA_{12} in a similar way. Finally, the computation of v and u are instances of $\mathcal{L}_{\mathbb{Z}}(r, ||A||, \max(||A||, ||b||))$.

The result now follows from Proposition 38 and 43. □

Corollary 45 ($R = \mathbb{Z}$) *Let nonsingular $A \in \mathbb{Z}^{n\times m}$ and $b \in \mathbb{Z}^{n\times 1}$ be given. The certified linear system solving problem with input (A, b) can be solved with an expected number of $O(nmr\,\mathsf{B}(d + \log m))$ bit operations, where r is the rank of A and d is a bound for both $\log||A||$ and $(\log||b||)/r$.*

PROOF. Let $\alpha = mM||A||$ and $\beta = \max(M, ||A||, ||b||)$. Then $\log \alpha = O(\log m + \log ||A||)$ and $\log \beta = O(r \log \alpha)$. Each instance of $\mathcal{L}(r, \alpha, \beta)$ can be solved in the allotted time using the algorithm supporting part 2 of Proposition 31. This requires knowing a $p \in \mathbb{Z}$ for which the input system remains nonsingular modulo p. Notice that, every time an instance of $\mathcal{L}_{\mathbb{Z}}(r, \alpha, \beta)$ needs to be solved in Algorithm *CertifiedSolver* or *SpecialMinimalSolution*, such a p has already been chosen and has bitlength bounded by $\log p = O(\log m + \log \log ||A||)$. This gives the estimate $O(nmr\,\mathsf{B}(d + \log m) + (n + m)\,\mathsf{B}(r(d + \log m)))$ for the expected number of required bit operations. The bound given in the statement of the corollary is actually a simplification, obtained using $\mathsf{B}(r(d + \log m)) = O(\mathsf{B}(r)\mathsf{B}(d + \log m))$, then $\mathsf{B}(r) = O(r^2)$. □

7.2 Complexity when $R = K[x]$

Any minor of A has degree bounded by $M := \min(n, m)||A||$. Choose T in step (1) differently depending on the size of K.

(Case 1: $\#K \geq 2M$) Choose T as explained in case 1 of §6.2 with $N := M$.

(Case 2a: $\#K < 2M$ and $\log_{\#K} \min(n, m) \leq ||A||$) Choose T as explained in case 2 of §6.2 with $N := M$.

(Case 2b: $\#K < 2M$ and $\log_{\#K} \min(n, m) > ||A||$) Construct an irreducible polynomial p of degree $2||A||$ (see Shoup (1994)) and compute $\bar{r}$ to be the rank of A mod $p \in (K[x]/(p))^{n \times m}$. By Lemma 46 we have $r \leq \bar{r} \leq 2r$ where r is the rank of A. Construct T as in case 2 of §6.2 with $N := \bar{r}||A||$.

In all cases, primes in T have degree bounded by $O(\log_{\#K} r + \log_{\#K} ||A||)$.

Lemma 46 *Let $A \in K[x]^{n \times m}$ and $p \in K[x]$ be irreducible. Let $\bar{r}$ be the rank of A mod $p \in (K[x]/(p))^{n \times m}$. If $\deg p > ||A||$, then the rank r of A over $K[x]$ satisfies $\bar{r} \leq r \leq \bar{r}/(1 - ||A||/\deg p)$.*

PROOF. The rank modulo a prime can only decrease so the claim $\bar{r} \leq r$ is clear. It remains to prove that $r \leq \bar{r}/(1 - ||A||/\deg p)$, which is equivalent to $r - \bar{r} \leq r||A||/\deg p$. If $r = \bar{r}$ the claim is true, so assume $\bar{r} < r$. Let $\mathrm{diag}(s_1, s_2, \ldots, s_r, 0, \ldots, 0)$ be the Smith form of A. Then $\sum_i \deg s_i \leq r||A||$, which is the maximum degree of any minor of A. Since p divides $s_{\bar{r}+1}$, we have $\deg s_{\bar{r}+1} \geq \deg p$. Since $s_1 | s_2 | \cdots | s_r$, we have $(r - \bar{r}) \deg p \leq \sum_i \deg s_i \leq r||A||$. The result follows. □

The proof of the next result is analogous to that of Proposition 44.

Proposition 47 ($R = K[x]$) *The expected cost of Algorithm* CertifiedSolver *is bounded by* $O(nm(\mathsf{MM}(r)/r^2)\,\mathsf{M}(d+t) + (n+m)\,\mathsf{B}(r(d+t)))$ *field operations, plus the cost of solving an expected* $O(1)$ *instances of* $\mathcal{L}_{K[x]}(r, ||A|| + t, \max(t, ||b||))$, *where* r *is the rank of* A, d *is a bound for both* $||A||$ *and* $||b||/r$, *and* $t = 0$ *if* $\#K$ *is infinite and* $t = \lfloor \log_{\#K}(3r||A||) \rfloor$ *otherwise.*

The proof of the next result is similar to that of Corollary 45, but now using Proposition 32 to bound the cost of solving the instances of $\mathcal{L}_{K[x]}(r, *, *)$.

Corollary 48 ($R = K[x]$) *Let nonsingular* $A \in K[x]^{n\times n}$ *and* $b \in K[x]^{n\times 1}$ *be given. The certified linear system solving problem with input* (A, b) *can be solved using an expected number of*

(1) $O(nmr\,\mathsf{B}(d+t))$, *or*
(2) $O(nm\,(\mathsf{MM}(r)/r^2)\,\mathsf{M}(d+t) + \mathsf{MM}(r)(\log r)\,\mathsf{B}(d+t) + (n+m)\,\mathsf{B}(r(d+t)))$

field operations, where r *is the rank of* A, d *is a bound for both* $||A||$ *and* $||b||/r$, *and* $t = 0$ *if* $\#K$ *is infinite and* $t = O(\log_{\#K} r)$ *otherwise.*

If we assume that $\mathsf{B}(r) = O(\mathsf{MM}(r)/r)$, which stipulates that if fast matrix multiplication techniques are used then fast polynomial arithmetic should be used also, then the bound in part 2 of Corollary 48 can be simplified to $O(nm(\mathsf{MM}(r)/r^2)(\log r)\mathsf{B}(d+t))$ field operations.

8 Shortest vector computation

We mention the notion of *minimal factor*. Let R be a principal ideal domain and F its fraction field. Let $Ax = b$ over R be consistent. The set of all $f \in F$ for which $Ax = fb$ admits a diophantine solution is a fractional ideal of R in F, that is an R–module $I \subseteq F$ such that $cI \subseteq R$ for some $c \in R \setminus \{0\}$ (see Lang (1986)). As in Section 2, we get a unique generator $f(A, b)$ for this fractional ideal — the set equals $f(A, b)R$. We call $f(A, b)$ the *minimal factor* of the system $Ax = b$. The vector $f(A, b)b$ is the shortest vector in the direction of b that is contained in the R–lattice spanned by the columns of A. It is not difficult to show that $f(A, b) = d(A, b/g)/g$, where g is the gcd of entries in b and $d(A, b/g)$ is the minimal denominator of $Ax = b/g$. Thus, $f(A, b)$ can be computed easily using the algorithms in this paper.

References

J. D. Dixon. Exact solution of linear equations using p-adic expansions. *Numer. Math.*, 40:137–141, 1982.

J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, 1999.

M. Giesbrecht. *Nearly Optimal Algorithms for Canonical Matrix Forms*. PhD thesis, University of Toronto, 1993.

M. Giesbrecht. Efficient parallel solution of sparse systems of linear Diophantine equations. In M. Hitz and E. Kaltofen, editors, *Second Int'l Symp. on Parallel Symbolic Computation: PASCO '97*, pages 1–10, New York, 1997. ACM Press.

M. Giesbrecht, A. Lobo, and B. D. Saunders. Certifying inconsistency of sparse linear systems. In O. Gloor, editor, *Proc. Int'l. Symp. on Symbolic and Algebraic Computation: ISSAC '98*, pages 113—119, New York, 1998. ACM Press.

G. H. Hardy and E. M. Wright. *An Introduction to the Theory of Numbers*. Oxford University Press, Oxford, 1979.

R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985.

D. E. Knuth. *The Art of Computer Programming, Vol.2, Seminumerical Algorithms*. Addison–Wesley, Reading MA, 2 edition, 1981.

S. Lang. *Algebraic number theory*. Springer-Verlag, New York, 1986.

R. Lidl and H. Niederreiter. *Finite Fields*. Addison-Wesley, Reading MA, 1983.

R. T. Moenck and J. H. Carter. Approximate algorithms to derive exact solutions to systems of linear equations. In Edward W. Ng, editor, *Proc. EUROSAM'79, LNCS 72*, pages 65–73, Heidelberg, 1979. Springer.

T. Mulders and A. Storjohann. Diophantine linear system solving. In S. Dooley, editor, *Proc. Int'l. Symp. on Symbolic and Algebraic Computation: ISSAC '99*, pages 181—188, New York, 1999. ACM Press.

T. Mulders and A. Storjohann. Rational solutions of singular linear systems. In C. Traverso, editor, *Proc. Int'l. Symp. on Symbolic and Algebraic Computation: ISSAC '00*, pages 242–249. ACM Press, New York, 2000.

J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Ill. J. Math.*, 6:64—94, 1962.

V. Shoup. Fast construction of irreducible polynomials over finite fields. *Journal of Symbolic Computation*, 17:371–391, 1994.

A. Storjohann. *Algorithms for Matrix Canonical Forms*. PhD thesis, Swiss Federal Institute of Technology, ETH–Zurich, 2000.

A. Storjohann. High–order lifting and integrality certification. *Journal of Symbolic Computation*, 36(3–4):613–648, 2003.

A. Storjohann and T. Mulders. Fast algorithms for linear algebra modulo N. In G. Bilardi, G. F. Italiano, A. Pietracaprina, and G. Pucci, editors, *Proc. of Sixth Ann. Europ. Symp. on Algorithms: ESA'98, LNCS 1461*, pages 139–150, 1998.

X. Wang and V. Y. Pan. Acceleration of euclidean algorithm and rational number reconstruction. *SIAM Journal of Computing*, 32(2):548–556, 2003.

D. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Trans.*

Inf. Theory, IT-32:54–62, 1986.